

1. A method for monitoring business transaction processing in an environment containing a number of component-based applications, wherein a first application emits a stream of events representative of a state transition or significant occurrence, the method comprising continuously:

5 detecting whether an event occurs in the first application, and for each event detected: a) capturing event data values corresponding to the event, wherein the event data values identify the application that generated the event and the time of the event generation; and b) collecting the event and the associated event data in a first event buffer; and

10 correlating events from the first event buffer into a correlation buffer, wherein the events in the correlation buffer are ordered according to the time of event generation;

2. A method for monitoring business transaction processing in an environment containing applications comprised of a plurality of components, 15 wherein a first application emits a stream of events representative of a state transition or a significant occurrence, the method comprising continuously:

detecting whether an event occurs in the first application, and for each event detected: a) capturing event data values corresponding to the event, wherein the event data values identify the application that generated the event and the time of the event generation; and b) collecting the event and the associated event data in a first event buffer; and

associating each event to at least one other event to create a merged event; and

25 creating a transaction from the merged event, the transaction comprising a start, an end, and a duration, wherein a transaction is a single, atomic operation performed on behalf of a particular user.

3. The method of claim 1 or 2 further comprising collecting a set of transactions to form a real-time transactional model of the business transaction processing.

30 4. The method of claim 1 or 2 wherein creating a transaction further comprises:

creating a model of the components of the application from the merged events; and

creating the transaction from the model and other merged events.

5. The method of claim 1 or 2 further comprising a second application

5 emitting a stream of events, the method further comprising:

detecting whether an event occurs in the second application, and for each event detected: a) capturing event data values corresponding to the event, wherein the event data values identify the application that generated the event and the time of the event generation; and b) collecting the event and the associated event data in a second event buffer; and

10 combining the events from the first event buffer and the second event buffer into a correlation buffer, wherein the events in the correlation buffer are ordered according to the time of event generation.

6. The method of claim 1 or 2 wherein creating a transaction further

15 comprises:

determining a begin event of the transaction;

determining a component employed by the transaction;

determining an end event of the transaction;

determining a transaction duration; and

20 determining a transaction name.

7. The method of claim 6 wherein determining a begin event further comprises determining whether an event is a root object method call, and if the event is a root object method call, assigning the root object method call as the begin event of the transaction.

25 8. The method of claim 6 wherein determining an end event further comprises determining whether an event is a method return event and if the event is a method return event, assigning the method return event as the end event of the transaction.

9. The method of claim 6 wherein collecting a set of transactions to form a

30 real-time transactional model of the business transaction processing further comprises:

- partitioning transactions into transaction sets based on the transaction name; and
- determining an active number of transactions and a completed number of transactions.
- 5    10.    The method of claim 1 or 2 further comprising:
- detecting a system event generated by an operating system, wherein the system event provides data descriptive of a process executing a transaction; and
- correlating the system event with the transaction.
11.    The method of claim 10 further including:
- 10    collecting a series of system events; and
- generating a performance curve of the system using the system events.
12.    The method of claim 11 further including correlating the set of transactions and the performance curve of the system to evaluate the business transaction processing.

PCT/US2019/032260